

V&V WITHIN REUSE-BASED SOFTWARE ENGINEERING

Reuse '96 Working Group

Edward A. Addy, Chair

30 July - 1 August 1996

Introduction

Verification and Validation (V&V) is used to increase the level of assurance of critical software, particularly that of safety-critical and mission-critical software. V&V is a systems engineering discipline that evaluates the software in a systems context, and is currently applied during the development of a specific application system. In order to bring the effectiveness of V&V to bear within reuse-based software engineering, V&V must be incorporated within the domain engineering process.

Participants in the working group developed an initial framework for performing V&V within reuse-based software engineering. This framework identified V&V tasks that could be performed in domain engineering, V&V tasks that could be performed in the transition from domain engineering to application engineering, and the impact of these tasks on application V&V activities. The group also considered the criteria and motivation for performing V&V in domain engineering.

The participants in the V&V Within Reuse-Based Software Engineering working group were:

- Edward Addy, NASA/WVU Software Research Laboratory (Chair)
- Susan Robinett, SRA / Army Reuse Center
- Michael Sabolish, MountainNet, Inc. / Research Intern
- Patrick Theeke, SAIC / ASSET

- Frances Van Scoy, WVU Department of Statistics and Computer Science

Verification and Validation in Traditional System Application Engineering

V&V is a set of activities performed in parallel with system development and designed to provide assurance that a software system meets the operational needs of the user. It ensures that the requirements for the system are correct, complete, and consistent, and that the life-cycle products correctly implement the system requirements.

The term *verification* refers to the process of determining whether or not the products of a given phase of the software development cycle fulfill the requirements established during the previous phase, while *validation* is the process of evaluating software at the end of the software development process to ensure compliance with software requirements [IEEE STD 729]. Verification is intended to ensure that the product is built correctly, while validation assures that the correct product is built.

While verification and validation have separate definitions, in practice the activities are merged into the process of V&V. This process evaluates software in a systems context, using a structured approach to analyze and test the software against system functions and against hardware, user and other software interfaces [Wallace and Fujii].